**[5 puntos] EJERCICIO 1: MODELADO LÓGICO RELACIONAL**

Realiza el paso a tablas en tercera forma normal (3FN).

Te mostramos un ejemplo de notación de modelado lógico que deberás cambiar por la de tu diagrama E-R:

**ID\_DIGITAL** (id, password, dir\_comp{tipo, nombre, piso, numero, localidad, circunscripción})

CP: id

Existe un atributo compuesto es dir\_comp, donde se especifica como restricción que se tiene que mantener la circunscripción según id digital, quitamos el atributo dir\_comp y lo movemos a una nueva tabla

**DIR\_ID** (circunscripción, tipo, nombre, piso, numero, localidad)

CP: circunscripcion

**ID\_DIGITAL**(id\_digital, circunscripción, password, dir\_comp, tipo, nombre, piso, numero, localidad)

CP: circunscripción → DIR\_ID

CP: id\_digital → ID\_DIGITAL

**ID\_DIGITAL** (id, password)

CP: id

**DIRECCION \_ID**( id circunscripción, tipo, nombre, piso, numero, localidad)

CP: id

CAj: id → ID\_DIGITAL{id}

CAj: circunscripción

UNICO: id

Ya está en 3FN, 1FN puesto que no hay atributos derivados, mutivaluados ni compuestos puesto que ya se ha creado tabla aparte por el atributo compuesto, 2FN porque tiene CP simple, todos los atributos dependen de la CP completa. 3FN no existen atributos con dependencias transitivas.

**Restricciones**:

(1) ATT derivado → No hay

(2) Pérdida semántica → Hay perdida semántica. Se necesita conocer la circunscripción según la ID\_DIGITAL.

Ya que en este caso la clave ajena que pasa a ser clave primaria en DIRECCION\_ID no puede repetirse, por tanto llevará la restricción de valor único.

(3) Cada Id digital puede emitir 3 votos como máximo.

La relación dispone se elimina en la normalización. Pero se necesita conocer la circunscripción según Id digital

////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////

**VOTANTE** (dni, id ,nom\_comp, teléfono,)

CP: dni

CAj: id → ID\_DIGITAL{id}

VNN: id → ID\_DIGITAL

Ya está en 3FN, 1FN puesto que no hay atributos derivados, mutivaluados ni compuestos, 2FN porque tiene CP simple, todos los atributos dependen de la CP completa. 3FN no existen atributos con dependencias transitivas.

**Restricciones**:

(1) ATT derivado → nom\_comp es un atributo compuesto, se podría descomponer siempre y cuando se especificara en el enunciado que fuera nombre + apellidos y marcado como tal, pero no es el caso.

(2) Pérdida semántica → No hay perdida semántica.

////////////////////////////////////////////////////////////////////////////////

**SENADOR** (dni , circ\_presenta, partido)

CP: dni

CAj: dni → VOTANTE {dni}

Tras la 1FN, también está en 2FN y también está en 3FN, no hay claves compuestas ni dependencias transitivas ni funcionales.

**SENADOR** (dni , circ\_presenta, partido)

CP: dni

CAj: dni → VOTANTE {dni}

Ya está en 3FN, 1FN puesto que no hay atributos derivados, mutivaluados ni compuestos, 2FN porque tiene CP simple, todos los atributos dependen de la CP completa. 3FN no existen atributos con dependencias transitivas.

**Restricciones**:

(1) ATT derivado → No hay.

(2) Pérdida semántica → Hay pérdida semántica, por la relación de herencia la especialización parcial por la que hereda la clave principal dni.

(4) En SENADOR los *partidos* válidos son: ASIR, DAM, DAW, PHT y PRO. Solo se almacenan las siglas en esta BD.

//////////////////////////////////////////////////////////////////////////////

**VOTA\_SEN**  ( id digital, circ\_presenta, num\_votos)

CP: id\_digital→ ID\_DIGITAL

CP: circ\_presenta→ SENADOR

CAj: circ\_presenta→ SENADOR {circ\_presenta}

CAj: id\_digital→ ID\_DIGITAL {id}

VNN: id\_digital→ ID\_DIGITAL

VNN: circ\_presenta→ SENADOR

Ya está en 3FN, pasa la 1FN y 2FN, no hay atributos multivaludaos, compuestos ni derivados. Y aunque hay clave compuesta especificada en enunciado, solo hay un campo no clave.

**Restricciones**:

(1) ATT derivado → No existen

(2) Pérdida semántica → Hay pérdida semántica. Deben coincidir la circunscripción de la Id digital y circ\_presenta del Senador.

El campo num\_votos almacena los votos que tiene el Senador, máximo 3 votos por id, este campo es imprescindible para tener este dato almacenado adecuadamente.

////////////////////////////////////////////////////////////////////////////////

**HORARIO** (día, hora\_inicio, hora\_fin)

CP: día, hora\_inicio, hora\_fin

UNICA: {día, hora\_inicio, hora\_fin}

Ya está en 3FN, 1FN puesto que no hay atributos derivados, mutivaluados ni compuestos, 2FN porque tiene CP simple, todos los atributos dependen de la CP completa. 3FN no existen atributos con dependencias transitivas.

**Restricciones**:

(1) ATT derivado → No hay

(2) Pérdida semántica → No hay perdida semántica.

(3) En tu caso, la clave primaria compuesta captura la unicidad de los horarios en función del día y las horas de inicio y fin.El *dia* debe estar entre el 3 y el 18 de marzo de 2025. La diferencia entre la *hora\_inicio* y la *hora\_fin* no puede ser superior a 2 horas. Obviamente la *hora\_inicio* debe ser anterior a la de fin.

////////////////////////////////////////////////////////////////////////////////

**INTERVENTOR** (dni , sueldo, id\_centro, supervisado\_por)

CP: dni → VOTANTE

CAj: dni → VOTANTE {dni}

CAj: id\_centro → CENTRO {id} B:C

CAj: supervisado\_por → INTERVENTOR {dni}

VNN: id\_centro

1FN: No existen atributos derivados, tiene atributos multivaluados que es sueldo, nos obliga a crear otra tabla y borrar el atributo en la tabla INTERVENTOR.

**SUELDO** (sueldo , dni\_interventor)

CP: dni → INTERVENTOR

Caj: sueldo → INTERVENTOR

**INTERVENTOR** (dni\_interv, supervisado\_por, id\_centro)

CP: dni\_interventor→ INTERVENTOR

CAj: dni → INTERVENTOR{ni}

CAj: supervisado\_por → INTERVENTOR {dni}

2FN: está en 1FN, tiene una CP simple por lo que todos sus atributos dependen de la CP completa, ya esta en 3FN puesto que no hay dependencias transitivas

**Restricciones**:

(1) ATT derivado → No hay

(2) Pérdida semántica → Hay pérdida semántica, el interventor se borra al eliminar su centro.

(3) Al ser una relación reflexiva aparece un nuevo atributo, en este caso supervisado\_por.

El interventor se borra al eliminar su Centro, es una debilidad de existencia.

////////////////////////////////////////////////////////////////////////////////

**CENTRO** (id\_centro, nombre\_centro, dirección,)

CP: id\_centro

Ya está en 3FN, 1FN puesto que no hay atributos derivados, mutivaluados ni compuestos, 2FN porque tiene CP simple, todos los atributos dependen de la CP completa. 3FN no existen atributos con dependencias transitivas.

**Restricciones**:

(1) ATT derivado → No hay.

(2) Pérdida semántica → No hay pérdida semántica.

(3) Restricción: el conjunto {*nombre\_centro*, *dirección*} es ÚNICO, no de forma independiente. Con esto quiere decir que tanto nombre\_centro como dirección serán campos únicos cuando se cree la tabla. Con esta restricción nos aseguramos a que tanto los nombres de los centros como las direcciones sean únicos en la tabla futura.

////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////

**PIDE\_C** (dni\_votante, dia, hora\_inicio, hora\_fin, id\_centro)

CP: { dni\_votante, dia, hora\_inicio, hora\_fin, id\_centro}

CAj: dni\_votante → VOTANTE {dni}

CAj: {dia, hora\_inicio, hora\_fin} → HORARIO

CAj: id\_centro → CENTRO {id\_centro}

Ya está en 3FN, 1FN puesto que no hay atributos derivados, mutivaluados ni compuestos, 2FN porque tiene CP simple, todos los atributos dependen de la CP completa. 3FN no existen atributos con dependencias transitivas.

**Restricciones**:

(1) ATT derivado → No hay atributos derivados

(2) Pérdida semántica → No hay perdidas semánticas

(3) Es la tabla resultante de la relación ternaria entre Centro, Horario y Votante, donde todas las claves primarias de cada entidad pasan a ser las claves primarias de la tabla resultante.

////////////////////////////////////////////////////////////////////////////////

**ACCION** (num\_reg, descripción, dni\_interventor)

CP: { num\_reg, dni\_interventor}

CAj: dni\_interventor→ INTERVENTOR {dni} B:C

Ya está en 3FN, 1FN puesto que no hay atributos derivados, mutivaluados ni compuestos, 2FN porque tiene CP simple, todos los atributos dependen de la CP completa. 3FN no existen atributos con dependencias transitivas.

**Restricciones**:

(1) ATT derivado → No hay atributos derivados

(2) Pérdida semántica → Hay pérdida semántica por la debilidad de identificación, por lo que se resuelve teniendo la clave primaria de la otra entidad INTERVENTOR como clave ajena, ya que necesita el dni del Interventor para ayudar a su identificación.

**[2 puntos] EJERCICIO 2: MODELADO FÍSICO DDL (CREACIÓN DE METADATOS)**

Indica las sentencias necesarias para incorporar las tablas del ejercicio anterior en MySQL. Recuerda tener cuidado con el orden de las instrucciones.

**IDIGITAL** (id, password)

CP: id

**DIRECCION \_ID**( id , circunscripción, tipo, nombre, piso, numero, localidad)

CP: id

CAj: id → ID\_DIGITAL{id}

CAj: circunscripción

UNICO: id

Creamos primero la base de datos

CREATE DATABASE IF NOT EXISTS eval2

CHARACTER SET utf8mb4 COLLATE utf8mb4\_es\_0900\_as\_cs;

USE eval2

CREATE TABLE id\_digital(

id INTEGER,

password CHAR(8),

CONSTRAINT idi\_cod\_pk PRIMARY KEY( id),

CONSTRAINT idi\_cod\_fk FOREIGN KEY (idigital) REFERENCES ID\_DIGITAL(id)

);

CREATE TABLE dirección\_id(

id INTEGER,

circunscripcion CHAR(9),

tipo VARCHAR(6),

nombre VARCHAR(30),

piso INTEGER,

numero INTEGER,

localidad VARCHAR(25),

CONSTRAINT dir\_id\_pk PRIMARY KEY (id),

CONSTRAINT dir\_circunscripcion\_fk FOREIGN KEY(circunscripcion) REFERENCES id\_digital(id),

CONSTRAINT dir\_id\_uk UNIQUE KEY (id)

);

**Restricciones**:

(1) Se necesita conocer la circunscripción según la ID\_DIGITAL.

(2) Ya que en este caso la clave ajena que pasa a ser clave primaria en DIRECCION\_ID no puede repetirse, por tanto, llevará la restricción de valor único.

(3) Cada Id digital puede emitir 3 votos como máximo.

La relación dispone se elimina en la normalización. Pero se necesita conocer la circunscripción según Id digital

////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////

**VOTANTE** (dni, id ,nom\_comp, teléfono,)

CP: dni

CAj: id → ID\_DIGITAL{id}

VNN: id → ID\_DIGITAL

CREATE TABLE votante(

dni VARCHAR(9),

id INTEGER NOT NULL,

nom\_comp VARCHAR(50),

teléfono INTEGER(9),

CONSTRAINT vot\_dni\_pk PRIMARY KEY(dni),

CONSTRAINT vot\_id\_fk FOREIGN KEY(id) REFERENCES id\_digital(id)

);

**Restricciones**:

(1) Se aplica valor no nulo a id, ya que es clave ajena que viene de CP de otra tabla.

////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////

**SENADOR** (dni , circ\_presenta, partido)

CP: dni

CAj: dni → VOTANTE {dni}

CREATE TABLE senador(

dni VARCHAR(9),

circ\_presenta INTEGER,

partido VARCHAR(4)NOT NULL CHECK(partido IN(‘ASIR’, ‘DAM’, ‘DAW’, ‘PHT’,‘PRO’)),

CONSTRAINT sen\_dni\_pk PRIMARY KEY(dni),

CONSTRAINT sen\_dni\_fk FOREIGN KEY(dni) REFERENCES votante(dni)

);

**Restricciones**:

(1) En senador los *partidos* válidos son: ASIR, DAM, DAW, PHT y PRO. Solo se almacenan las siglas en esta BD.

////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////

**VOTA\_SEN** ( id digital, circ\_presenta, num\_votos)

CP: id\_digital→ ID\_DIGITAL

CP: circ\_presenta→ SENADOR

CAj: circ\_presenta→ SENADOR {circ\_presenta}

CAj: id\_digital→ ID\_DIGITAL {id}

VNN: id\_digital→ ID\_DIGITAL

VNN: circ\_presenta→ SENADOR

CREATE TABLE vota\_sen(

id\_digital INTEGER,

circ\_presenta INTEGER,

num\_votos INTEGER NOT NULL CHECK (num\_votos BETWEEN 0 AND 3),

CONSTRAINT vot\_id\_ci\_pk PRIMARY KEY (id\_digital, circ\_presenta),

CONSTRAINT vot\_id\_fk FOREIGN KEY (id\_digital) REFERENCES id\_digital(id),

CONSTRAINT vot\_cir\_fk FOREIGN KEY (circ\_presenta) REFERENCES senador(circ\_presenta),

CONSTRAINT CHECK (EXISTS (SELECT 1 FROM senador WHERE senador.circ\_presenta = vota\_sen.circ\_presenta AND senador.id = vota\_sen.id\_digital))

);

**Restricciones**:

(1)En *Vota\_Sen* la *circunscripción* de la ID\_DIGITAL y *circ\_presenta* del SENADOR deben coincidir.

(2)El campo num\_votos almacena los votos que tiene el Senador, máximo 3 votos por id, este campo es imprescindible para tener este dato almacenado adecuadamente

////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////

**SUELDO** (sueldo, dni\_interventor)

CP: dni → INTERVENTOR

Caj: sueldo → INTERVENTOR

CREATE TABLE sueldo (

sueldo INTEGER(6),

dni\_interventor CHAR(9),

CONSTRAINT sue\_dni\_pk PRIMARY KEY (dni\_interventor) REFERENCES interventor(dni\_interventor),

CONSTRAINT sue\_sueldo\_fk FOREIGN KEY (sueldo) REFERENCES interventor(sueldo)

);

**INTERVENTOR** (dni\_interv, supervisado\_por, id\_centro)

CP: dni\_interventor→ INTERVENTOR

CAj: dni → INTERVENTOR{ni}

CAj: supervisado\_por → INTERVENTOR {dni}

VNN: supervisado\_por → INTERVENTOR {dni}

CREATE TABLE interventor (

dni\_interv CHAR (9),

supervisado\_por VARCHAR(50) NOT NULL,

id\_centro INTEGER,

CONSTRAINT int\_dn\_pk PRIMARY KEY(dni\_interv)

CONSTRAINT int\_dn\_fk FOREIGN KEY(dni\_interv) REFERENCES SUELDO(dni\_interventor) ON DELETE CASCADE,

CONSTRAINT int\_sup\_fk FOREIGN KEY(supervisado\_por) REFERENCES INTERVENTOR(dni\_interv),

CONSTRAINT int\_id\_c\_fk FOREIGN KEY(id\_centro) REFERENCES centro(id\_centro) ON DELETE CASCADE

CONSTRAINT int\_dn\_uk UNIQUE (dni\_interv),

);

**Restricciones**:

(1) Hay pérdida semántica, el interventor se borra al eliminar su centro.

(2) El interventor se borra al eliminar su Centro, es una debilidad de existencia

////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////

**HORARIO** (día, hora\_inicio, hora\_fin)

CP: día, hora\_inicio, hora\_fin

UNICA: {día, hora\_inicio, hora\_fin}

CREATE TABLE horario(

día DATE,

hora\_inicio TIME,

hora\_fin TIME,

CONSTRAINT hor\_di\_hi\_hf\_pk PRIMARY KEY(día, hora\_inicio, hora\_fin),

CONSTRAINT hor\_dia\_ck CHECK (dia BETWEEN '2025-03-03' AND '2025-03-18'),

CONSTRAINT hor\_dia\_ck CHECK (hora\_inicio < hora\_fin),

CONSTRAINT hor\_hi\_hf\_ck CHECK (TIMESTAMPDIFF(HOUR, hora\_inicio, hora\_fin) <= 2)

);

**Restricciones**:

(1)El *día* debe estar entre el 3 y el 18 de marzo de 2025. La diferencia entre la *hora\_inicio* y la *hora\_fin* no puede ser superior a 2 horas. Obviamente la *hora\_inicio* debe ser anterior a la de fin.

////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////

**CENTRO** (id\_centro, nombre\_centro, dirección,)

CP: id\_centro

CREATE TABLE centro(

id\_centro INT,

nombre\_centro VARCHAR(50),

direccion VARCHAR(80),

CONSTRAINT cen\_id\_pk PRIMARY KEY,

CONSTRAINT cen\_no\_di\_uk UNIQUE (nombre\_centro, dirección)

);

**Restricciones**:

(1) Restricción: el conjunto {*nombre\_centro*, *dirección*} es ÚNICO, no de forma independiente. Con esto quiere decir que tanto nombre\_centro como dirección serán campos únicos cuando se cree la tabla. Con esta restricción nos aseguramos a que tanto los nombres de los centros como las direcciones sean únicos en la tabla futura.

////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////

**PIDE\_C** (dni\_votante, dia, hora\_inicio, hora\_fin, id\_centro)

CP: { dni\_votante, dia, hora\_inicio, hora\_fin, id\_centro}

CAj: dni\_votante → VOTANTE {dni}

CAj: {dia, hora\_inicio, hora\_fin} → HORARIO

CAj: id\_centro → CENTRO {id\_centro

CREATE TABLE pide\_C(

dni\_votante VARCHAR(9),

dia DATE,

hora\_inicio TIME,

hora\_fin TIME,

id\_centro CHAR(9),

CONSTRAINT pid\_dn\_di\_hi\_hf\_pk PRIMARY KEY (dni\_votante, dia, hora\_inicio, hora\_fin, id\_centro),

CONSTRAINT pid\_dni\_fk FOREIGN KEY (dni\_votante) REFERENCES votante(dni),

CONSTRAINT pid\_di\_hi\_hf\_fk FOREIGN KEY (dia, hora\_inicio, hora\_fin) REFERENCES horario(dia, hora\_inicio, hora\_fin),

CONSTRAINT pid\_id\_fk FOREIGN KEY (id\_centro) REFERENCES centro(id\_centro)

);

////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////

**ACCION** (num\_reg, descripción, dni\_interventor)

CP: { num\_reg, dni\_interventor}

CAj: dni\_interventor→ INTERVENTOR {dni} B:C

CREATE TABLE accion (

num\_reg INTEGER,

descripción VARCHAR(280),

dni\_interventor VARCHAR(9),

CONSTRAINT acc\_nu\_dn\_pk PRIMARY KEY (num\_reg, dni\_interventor),

CONSTRAINT acc\_dn\_fk FOREIGN KEY (dni\_interventor) REFERENCES interventor(dni\_interv) ON DELETE CASCADE

);

**Restricciones**:

(1)Pérdida semántica → Hay pérdida semántica por la debilidad de identificación, por lo que se resuelve teniendo la clave primaria de la otra entidad INTERVENTOR como clave ajena, ya que necesita el dni del Interventor para ayudar a identificación

**[1 punto] EJERCICIO 3: MODELADO FÍSICO DDL (MODIFICACIÓN DE METADATOS)**

Indica las sentencias necesarias para realizar estas modificaciones en MySQL. Recuerda tener cuidado con el orden de las instrucciones.

* Eliminar la relación *Dispone* y cualquier referencia al votante en ID\_DIGITAL a excepción de la *circunscripcion*.

//Eliminamos la relación Dispone y comprobamos

ALTER TABLE dispone

DROP FOREIGN KEY dis\_id\_\_fk;

DESC dispone;

//Eliminamos cualquier referencia al votante en ID\_DIGITAL excepto la circunscripción y comprobamos

ALTER TABLE id\_digital

DROP FOREIGN KEY id\_id\_cod\_fk;

DESC id\_digital;

* Ahora todos los INTERVENTORES son supervisados, la participación pasa de (0,1) a (1,1).

//Dentro de la tabla interventor modificamos el valor de supervisado\_por a un valor no nulo y después comprobamos

ALTER TABLE interventor

MODIFY COLUMN supervisado\_por INT NOT NULL;

DESC interventor;

* Si se elimina una ID\_DIGITAL se borran sus votos también.

//Primero eliminamos la fila en concreto con DROP FOREIGN luego agregamos(añadimos con su correspondiente CONSTRAINT a la clave ajena) ON DELETE CASCADE a la clave foránea en la tabla VOTA\_SEN, ya que no queremos eliminar una tabla sino lo que queremos es eliminar un id\_digital y sus votos, queremos eliminar una fila de una tabla y a continuación on update para que se actualize correctamente ya que se ha eliminado una fila. Después comprobamos

ALTER TABLE vota\_sen

DROP FOREIGN KEY vot\_id\_\_fk;

ALTER TABLE vota\_sen

ADD CONSTRAINT vot\_id\_\_fk;

FOREIGN KEY (id\_digital) REFERENCES ID\_DIGITAL(id) ON DELETE CASCADE ON UPDATE CASCADE;

DESC vota\_sena;

* El sueldo en INTERVENTOR tiene un valor por defecto de 5000.

//Agregamos el valor por defecto de 5000 al sueldo del interventor y comprobamos

ALTER TABLE interventor

MODIFY COLUMN sueldo\_interventor DEFAULT 5000;

DESC interventor;

* El *nombre\_centro* y *direccion* dejan de ser únicos en conjunto, ahora solo la *direccion* del centro es única.

//Eliminamos restricción UNIQUE en el conjunto{nombre\_centro,dirección} en la tabla CENTRO.

ALTER TABLE centro

DROP INDEX unique\_nombre\_centro\_direccion;

//Agregamos restricción UNIQUE solo a direccion en la tabla CENTRO y luego

Comprobamos.

ALTER TABLE centro

ADD UNIQUE (direccion);

DESC centro;

**[2 puntos] EJERCICIO 4: MODELADO FÍSICO DML (MANIPULACIÓN DE DATOS)**

Indica las sentencias necesarias para realizar estas modificaciones en MySQL. Recuerda tener cuidado con el orden de las instrucciones.

* **Operación 1. Crea una ID\_DIGITAL con los datos que quieras en la *circunscripción* "C.Valenciana" que haya votado al SENADOR del "PHT".**

//Creamos id digital llamado 987654

INSERT INTO id\_digital (id, circunscripcion, password)

VALUES ('987654', 'C.Valenciana', 'mi\_password');

* **Operación 2. Crea 2 SENADORES con los siguientes datos:**

***dni* "11111111A", *partido* "PRO" *circ\_presenta* "C.Valenciana".**

// Se añaden datos dentro de la tabla senador

INSERT INTO senador (dni, partido, circ\_presenta)

VALUES ('11111111A', 'PRO', 'C.Valenciana');

***dni* "22222222B", *partido* "PHT" *circ\_presenta* "C.Valenciana".**

// se añaden datos dentro de la tabla senador

INSERT INTO senador (dni, partido, circ\_presenta)

VALUES ('22222222B', 'PHT', 'C.Valenciana');

* **Operación 3. Crea 1 INTERVENTOR con los siguientes datos: dni "12345678C", nom\_comp "Juan Carlos Gurtelero Eresísimos", sueldo por defecto, supervisado por él mismo.**

//En esta sentencia utilizamos el campo sueldo DEFAULT tomando el valor por defecto que habíamos definido anteriormente

INSERT INTO interventor (dni\_interv, nom\_comp, sueldo, supervisado\_por)

VALUES ('12345678C', 'Juan Carlos Gurtelero Eresísimos', DEFAULT, '12345678C');

**Pidió cita para votar el día 13 de marzo entre las 15:00 y las 17:30.**

//Registramos la cita para votar asegurándolo en un id\_centro valido, ajustando el día, la fecha y la hora solicitada

INSERT INTO pide\_C (dni\_votante, dia, hora\_inicio, hora\_fin, id\_centro)

VALUES ('12345678C', '2025-03-13', '15:00:00', '17:30:00', 'mi\_id\_centro');

**Registra una ACCION con *descripcion* "He votado por los jubilados de la residencia La Viña".**

//Resgistramos la acción de manera secuencial con su descripción

INSERT INTO accion (num\_reg, descripcion, dni\_interventor)

VALUES (1, 'He votado por los jubilados de la residencia La Viña', '12345678C');